New Identity-Based Authenticated Key Agreement Protocols from Pairings

Shengbao Wang and Zhenfu Cao

Department of Computer Science and Engineering,
Shanghai Jiao Tong University,
800 Dongchuan Road, Shanghai 200240, P.R. China
{shengbao-wang,cao-zf}@cs.sjtu.edu.cn
http://tdt.sjtu.edu.cn

Abstract. Key agreement protocols are essential for secure communications in open and distributed environments. As identity-based cryptography has become extremely fashionable in the last few years, many identity-based key agreement protocol have emerged, among which most of them are based on pairings. In this paper, we present two new such protocols. Our constructions use the ideas from the newly proposed identity-based encryption scheme of Gentry [15], which was proved to be secure without the random oracle. Our protocols are expected to be proved secure in the standard model (without random oracles).

Keywords: identity-based cryptography, authenticated key agreement, bilinear pairings

1 Introduction

Key agreement protocols are of fundamental importance for communications between two parties over an insecure network. Informally, *authenticated key agreement* (AK) protocols not only allow parties to compute a *session key* known only to them but also ensure authenticity of the parties [6]. This secret session key can then be used to provide privacy and data integrity during subsequent sessions.

In 1976, Diffie and Hellman [12] proposed the first key agreement protocol in their seminal paper that also marked the birth of public-key cryptography (PKC). If in a protocol one party is assured that no other party aside from the designated party (or parties) may gain access to the particular established secret key, then the key agreement protocol is said to provide *implicit key authentication* (IKA). An authenticated key agreement protocol provides mutual IKA between (or among) parties. A key agreement protocol provides key confirmation (of B to A) if A is assured that B actually possesses the session key. An AK protocol that provides mutual key confirmation is called an authenticated key agreement with key confirmation protocol (or an AKC protocol). Key agreement protocols employ private or public-key cryptography. In this paper, we shall only consider two-party key agreement protocols in the public-key setting.

A Generic Design Strategy. Signature-/encryption-less key agreement protocols have numerous advantages, and namely from the efficiency point of view. They are thus well-suited for some constrained environments [18]. To our knowledge, the common strategy of constructing an signature-/encryption-less *authenticated* key agreement protocol using the idea behind the ElGamal [13] public key encryption scheme was first suggested by Matsumoto, Takashima and Imai in 1986 [21]. The authors designed several authenticated Diffie-Hellman key agreement protocols (i.e., to provide the original Diffie-Hellman protocol with key authentication), which is well-known as the MTI key agreement family. In particular, the MTI/A0 protocol is derived from the ElGamal encryption in such a way that we name it as a mutual "`Encryption and Decryption`" mechanism (refer to Section 3 for details). Similar but later proposals are the protocol of Goss [16], KEA [22] authenticated key agreement designed by NSA in 1994 (declassified in 1998) and Protocol 4 from Blake-Wilson et al. [5].

Although this strategy has already been used for many times in the research community, for the first time, we make it explicit, and show its effectiveness again by proposing two new ID-based AK protocols using pairings.

Related Work. The idea of *identity (ID)-based cryptography* was first introduced by Shamir in 1984 [24] . The basic idea behind an ID-based cryptosystem is that end users can choose an arbitrary string, for example their email addresses or other online identifiers, as their public key. This eliminates much of the overhead associated with key management [19]. In 2001, Boneh and Franklin [2] gave the first fully functional solution for ID-based encryption (IBE), which is an variant of the ElGamal [13] encryption scheme, using the pairing on elliptic curves. Since then, abundant ID-based AK protocols with or without signatures using pairings have also been suggested (e.g., [25,26,11,23,10]).

Smart's protocol [25] is the first ID-based AK protocol using pairings. It is based on the idea of the Boneh-Franklin IBE scheme, without using any encryption or digital signature scheme. We note that Smart's ID-based protocol was exactly designed employing the common strategy from [21].

Our Contributions. In this paper, based on the IBE system of Gentry [15] (which was presented at Eurocrypt 2006), we put forward two ID-based AK protocols.

Recent and independent work of [27] gives a 3-pass ID-based AKC protocol which also uses ideas from Gentry's IBE system. However, we point out that their protocol does *not* comply with the design strategy we mentioned above and, unfortunately, as we find out that it is vulnerable to a key-compromise attack. Our successful K-CI attack invalidate their formal security proof.

Our contributions in this paper are:

- An detailed refinement and description of an effective design strategy for ID-based authenticated key agreement protocols using ElGamal-type ID-based encryption systems.
- Two new ID-based AK protocols that can be instantiated with or without PKG forward secrecy (also known as *master-key* forward secrecy).
- A key-compromise impersonation attack on a newly proposed ID-based AKC protocol.

Paper Organization. The rest of this paper is structured as follows. In the next section, we give the necessary technical backgrounds. Section 3 presents the effective design strategy for ID-based authenticated key agreement protocols. Section 4 reviews Gentry's ID-based encryption scheme. In Section 5, we put forward our new protocols which can be expected to be proven secure without random oracles. We draw some conclusions in Section 6. Finally, our K-CI attack on the protocol from [27] is given in Appendix A.

2 Technical Backgrounds

2.1 Security Requirements

In the past, some desired security attributes for AK(C) protocols have been identified in [6,20,7]. We briefly explain the security attributes as follows (refer to [6,20] for more detailed discussions):

- **Known-key secrecy.** Suppose an established session key between two entities is disclosed, the adversary is unable to learn other established session keys.
- **Perfect forward secrecy (PFS).** If both long-term secret keys of two entities (i.e. the protocol principals) are disclosed, the adversary is unable to derive old session keys established by that two entities.
- **Key-compromise impersonation (K-CI) resilience.** Assume that entities A and B are two principals. Suppose $A's$ secret key is disclosed. Obviously, an adversary who knows this secret key can impersonate A to other entities (e.g. B). However, it is desired that this disclosure does not allow the adversary to impersonate other entities (e.g. B) to A.

– **Unknown key-share (UK-S) resilience.** Entity A cannot be coerced into sharing a key with entity B without A's knowledge, i.e., when A believes that the key is shared with some entity $C \neq B$, and B (correctly) believes the key is shared with A.
– **No key control.** Neither the two protocol principals (A and B) can predetermine any portion of the shared session key being established between them.

The main desirable *performance attributes* include low computational overhead, a minimal number of passes (the number of messages exchanged in a run of the protocol), and low communication overhead (total number of bits transmitted).

2.2 Bilinear Pairings

In this section, we describe in a more general format the basic definition and properties of the pairing: more details can be found in [2].

Let $\mathbb{G}_1$ be a cyclic additive group generated by an element P, whose order is a prime p, and $\mathbb{G}_2$ be a cyclic multiplicative group of the same prime order p. We assume that the discrete logarithm problem (DLP) in both $\mathbb{G}_1$ and $\mathbb{G}_2$ are hard.

Definition 1. *An admissible pairing e is a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, which satisfies the following three properties:*

1. Bilinear*: If $P, Q \in \mathbb{G}_1$ and $a, b \in Z_p^*$, then $e(aP, bQ) = e(P, Q)^{ab}$;*
2. Non-degenerate*: There exists a $P \in \mathbb{G}_1$ such that $e(P, P) \neq 1$;*
3. Computable*: If $P, Q \in \mathbb{G}_1$, one can compute $e(P, Q) \in \mathbb{G}_2$ in polynomial time.*

2.3 Complexity Assumptions

The security of Gentry's IBE system [15] is based on a complexity assumption that they call *the truncated augmented bilinear Diffie-Hellman exponent assumption* (the truncated q-ABDHE). We recall the truncated q-ABDHE problem as follows (refer to [15] for detailed description): Given a vector of $q + 3$ elements

$$(P', \alpha^{q+2}P', P, \alpha P, \alpha^2 P, ..., \alpha^q P) \in \mathbb{G}_1^{q+3}$$

as input, an algorithm $\mathcal{A}$ that outputs $b \in \{0, 1\}$ has advantage ϵ in solving the truncated decision q-ABDHE if

$$\begin{aligned} \mid Pr[\mathcal{A}(P', \alpha^2 P, P, \alpha P, ..., \alpha^q P, e(\alpha^{q+1}P,\ P')] = 0 \\ -Pr[\mathcal{A}(P', \alpha^2 P, P, \alpha P, ..., \alpha^q P, Z) = 0] \mid \geq \epsilon \end{aligned}$$

where the probability is over the random choice of generators P, $P' \in \mathbb{G}_1$, the random choice of $\alpha \in \mathbb{Z}_p$, the random choice of $Z \in \mathbb{G}_2$, and the random bits consumed by $\mathcal{A}$.

Definition 2. *We say that the truncated (decision) (t, ϵ, q)-ABDHE assumption holds in $\mathbb{G}_1$ if no t-time algorithm has advantage at least ϵ in solving the truncated (decision) q-ABDHE problem in $\mathbb{G}_1$.*

3 A Design Strategy for ID-Based AK Protocols

Here we describe in detail a design strategy for ID-based authenticated key agreement protocols, based on any ElGamal-type ID-based encryption system.

To illustrate the ideas behind the above mentioned strategy, we first recall the so-called ElGamal one-pass *unilateral* key agreement protocol that was first given in Chap. 12 of [20]. We quote it as follows.

"ElGamal key agreement is a Diffie-Hellman variant providing a one-pass protocol with unilateral key authentication (of the intended recipient to the originator), provided the public key of the recipient is known to the originator a priori. While related to ElGamal encryption, the protocol is more simply Diffie-Hellman key agreement wherein the public exponential of the recipient is fixed and has verifiable authenticity (e.g., is embedded in a certificate)."

Informally, the protocol proceeds as follows. The sender A forms a *shared* secret using her random input r_A in combination with B's long-term public key y_B by computing $K = y_B^{r_A}$. On receipt of the ephemeral public key $T_A = r_A P$, the receiver B is able to reconstruct the session key $K = T_A^{x_B}$.

The two-pass MTI/A0 protocol can be seen as a parallel execution of ElGamal one-pass key agreement protocol. It yields session keys with *mutual* (implicit) key authentication against passive attacks. As in ElGamal one-pass key agreement, A sends to B a single message, resulting in the shared key K. B independently initiates an analogous protocol with A, resulting in the shared key K'. A and B then output the KK' as their agreed session key.

Note that although the original MTI/A0 protocol is of certain security weaknesses, e.g., it doesn't provide perfect forward secrecy and is vulnerable to some active attacks such as unknown key-share attacks [7], triangle attacks [8], the elegant idea behind it is very useful in Diffie-Hellman authenticated key agreement protocol design.

The above idea can be applied to the design of pairing-based AK protocols. Inspired by the above design strategy, Al-Riyami and Paterson [1] gave the first certificateless authenticated key agreement protocol based on their certificateless public key encryption (CL-PKE) scheme (also appears in [1]) in 2003. Similarly, Wang and Cao [28] proposed the first certificate-based authenticated key agreement protocol in 2004, using the ideas from the certificate-based encryption (CBE) scheme of Gentry [14].

So far, we are ready to define a general framework, named as *the general MTI/A0 protocol* (GMP), for the design of ID-based AK protocols based on an ElGamal-type IBE scheme.

Definition 3. General MTI/A0 Protocol (GMP). *Suppose we have an ElGamal-type IBE scheme and two users (Alice and Bob) want to agree on a shared session key. They does the following:*

1. Alice (Bob) firstly generates her (his) ephemeral private key, then computes her (his) ephemeral public key PK_{eph} using the public parameters of the system, finally she (he) sends PK_{eph} to Bob (Alice).
2. Alice (Bob) uses the ephemeral private key generated in Step 1 to compute *the ElGamal encryption session key* K_{En}.
3. After receiving the ephemeral public key, they each calculate *the ElGamal decryption session key* K_{De}, using their own long-term private keys.
4. Alice (Bob) computes her (his) final session key sk as follows. (Where the symbol "$*$" stands for such operations as muliplication, addition, or bitwise XOR.)

$$sk = K_{En} * K_{De}.$$

Remark 1. The two users are able to sucessfully establish a shared session key after the above GMP.

4 Gentry's IBE Scheme

In this section, we review the first construction of Gentry from [15], which is a chosen-plaintext secure ElGamal-type IBE scheme (proved in the standard model).

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order p, and let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be the bilinear pairing. The IBE system works as follows.

Setup: The private key generator (PKG) chooses two random generators $P, Q \in \mathbb{G}_1$ and a random $\alpha \in \mathbb{Z}_p$, calculates $P_1 = \alpha P \in \mathbb{G}_1$. It sets the public *params* as $< P, P_1, Q >$ and the *master-key* as α.
Key Generation: To generate a private key for identity $ID \in \mathbb{Z}_p$, the PKG generates a random $r_{ID} \in \mathbb{Z}_p$, and outputs the private key as $d_{ID} =< r_{ID}, h_{ID} >$, where $h_{ID} = (\alpha - ID)^{-1} \cdot ((-r_{ID}) \cdot P + Q)$. (The PKG ensures that $ID \neq \alpha$ and it always assigns identical r_{ID} for a given identity ID.)
Encryption: The sender picks randomly a $s \in \mathbb{Z}_P$, using the receiver's identity ID, sets the ciphertext to be (to encrypt message $m \in \mathbb{G}_2$)

$$C = (\ s \cdot (P_1 + (-ID) \cdot P),\ e(P,P)^s,\ m \bullet e(P,Q)^{-s}\ ).$$

Decryption: To decrypt ciphertext $C = (u, v, w)$, the decrypter of the identity ID computes

$$m = w \bullet e(u,\ h_{ID}) \bullet v^{r_{ID}}.$$

Consistence: The recipient can correctly decrypt C to get m since

$$\begin{aligned} & e(u,\ h_{ID}) \bullet v^{r_{ID}} \\ = & e(s(\alpha - ID) \cdot P,\ (\alpha - ID)^{-1} \cdot Q + (-r_{ID})(\alpha - ID)^{-1} \cdot P) \bullet e(P,P)^{sr_{ID}} \\ = & e(P,Q)^s. \end{aligned}$$

5 New ID-Based Authenticated Key Agreement Protocols

In this section, we propose two new ID-based authenticated key agreement based on Gentry's IBE scheme (refer to 4). Naturally, we use the design strategy that we described in Section 3.

Our first protocol (named as Protocol I) does not provide PKG forward secrecy (or master-key forward secrecy), i.e., when the maser key α of the PKG is compromised, an adversary who gets it can recover all the users' past session keys. Equivalently, this means that the PKG can *escrow* all the session keys (refer to [11,19] for more details). Clearly, for any ID-based key agreement protocol, perfect forward secrecy (PFS) is implied by the PKG forward secrecy. Whereas, our second protocol (named as II) provides the PKG forward secrecy. Both of our protocols are two-pass protocol with *mutual* implicit key authentication (IKA). We note that it is readily to extend our protocols into 3-pass AKC protocols, using the common method given in [5,11].

5.1 A Protocol without PKG Forward Secrecy: Protocol I

As with all the other ID-based AK protocols we assume the existence of a PKG that is responsible for the creation and secure distribution of users' private keys.

Protocol I consists of three stages, i.e. `Setup`, `Key Generation` and `Key Agreement`. The `Setup` and `Key Generation` stages are identical to that of Gentry's IBE scheme [15].

Suppose two principals Alice and Bob are about to agree on a session key (we denote their identity as ID_A and ID_B, respectively), we follow previous notations and hereafter, let $g_{ID} = P_1 + (-ID_B) \cdot P$ and $g_T = e(P,P)$. The `Key Agreement` stage is as follows.

Key Agreement. To establish a shared session key, Alice and Bob each firstly generate an ephemeral private key (say x and $y \in \mathbb{Z}_p$), and compute the corresponding ephemeral public keys $T_{11} = x \cdot g_{ID_B}$, $T_{12} = g_T^x$ and $T_{21} = y \cdot g_{ID_A}$, $T_{22} = g_T^y$. They then exchange $T_1 = T_{11}||T_{12}$ and $T_2 = T_{21}||T_{22}$ as described in Figure 1 (where the symbol $''||''$ denotes concatenation).

After the message exchange, the two users do the following:

1. Alice computes the shared secret K_{AB} as follows:

$$K_{AB} = [e(T_{21},\ h_{ID_A}) \bullet (T_{22})^{r_{ID_A}}] \bullet e(P,\ Q)^x.$$

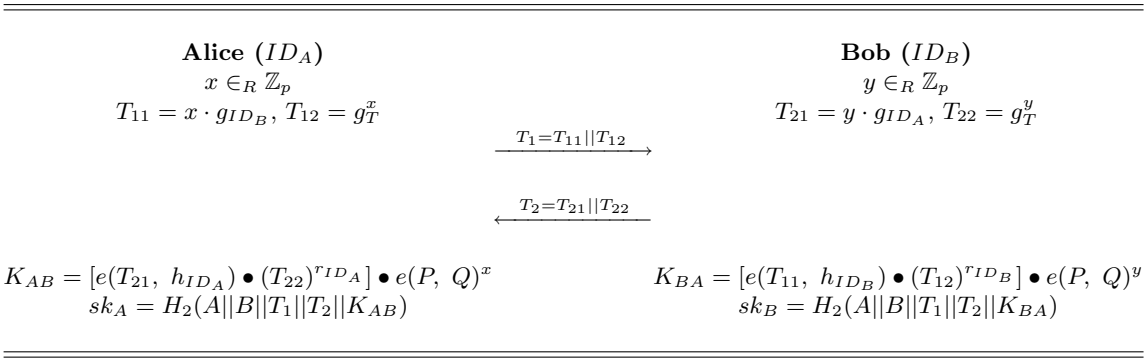

Fig. 1. Protocol I

2. Bob computes the shared secret K_{BA} as follows:

$$K_{BA} = [e(T_{11},\ h_{ID_B}) \bullet (T_{12})^{r_{ID_B}}] \bullet e(P,\ Q)^y.$$

Protocol Correctness: By the bilinearity of the pairing, we can easily get the following equations:

$$\begin{aligned}
K_{AB} &= e(T_{21},\ h_{ID_A}) \bullet (T_{22})^{r_{ID_A}} \bullet e(P,\ Q)^x \\
&= e(y \cdot g_{ID_A},\ (\alpha - ID_A)^{-1} \cdot (-r_{ID_A} \cdot P + Q)) \bullet (g_T^y)^{r_{ID_A}} \bullet e(P,\ Q)^x \\
&= e(y(\alpha - ID_A) \cdot P,\ (\alpha - ID_A)^{-1} \cdot (-r_{ID_A} \cdot P + Q)) \bullet (g_T^y)^{r_{ID_A}} \bullet e(P,\ Q)^x \\
&= e(y \cdot P,\ (-r_{ID_A} \cdot P + Q)) \bullet (g_T^y)^{r_{ID}} \bullet e(P,\ Q)^x \\
&= e(y \cdot P,\ (-r_{ID_A} \cdot P + Q)) \bullet (e(P,\ P)^y)^{r_{ID_A}} \bullet e(P,\ Q)^x \\
&= e(y \cdot P,\ (-r_{ID_A} \cdot P + Q)) \bullet e(y \cdot P,\ r_{ID_A} \cdot P) \bullet e(P,\ Q)^x \\
&= e(y \cdot P,\ Q) \bullet e(P,\ Q)^x \\
&= e(P, Q)^{x+y}.
\end{aligned}$$

Analogously, we can get $K_{BA} = e(P, Q)^{x+y}$. Thus, the two secret keys computed by Alice and Bob (K_{AB} and K_{BA}) are equal to each other, i.e., the two users successfully established a shared secret $K = K_{AB} = K_{BA}$ after running an instance of the protocol. The final shared secret session key is then $sk = H_2(A||B||T_A||T_B||K)$, where $H_2 : \{0,1\}^* \rightarrow \{0,1\}^k$ is a *key derivation function* (in which $k = |sk|$). Note here we include the transcript (T_1 and T_2) in the key derivation function to resist the potential *key replicating attack* (whereby an adversary is somehow able to manipulate the shared secret K using his own contributions while he still does not know the value of K) [9].

Our protocol is *role symmetric*, which means that each party performing the same operations. As has been pointed out in [5,17], for the sake of formal security proof, it would be better to introduce *asymmetry* into the session key generation, e.g., we can set the shared key K as $e(P, Q)^x||e(P, Q)^y$, instead of $e(P, Q)^x \bullet e(P, Q)^y$.

Remark 2. We argue that Protocol I achieves all the desired security attributes for an AK protocol, except for perfect forward secrecy as well as PKG forward secrecy. We let the formal security proof as future work.

Remark 3. Thanks to Gentry's IBE scheme, Protocol I can be expected to be proven secure in the model of Bellare and Rogaway *without random oracles*.

It is easy to find out that Protocol I does not achieve PKG forward secrecy, since with the knowledge with all the two users' private keys, an adversary is also able to calculate $e(P, Q)^x$

and $e(P,Q)^y$ (denoted as K_{En} and K_{De} in the definition of GMP of Section 3) using the publicly resubmitted data, thus results in the compromise of all the previous established session keys.

5.2 A Protocol with PKG Forward Secrecy: Protocol II

Similar to the idea used in [11], we calculate an extra *Diffie-Hellman* shared key from the two participants' *ephemeral* contributions. Unlike the protocols in [11], Protocol II does not bring additional communication cost. We now introduce Protocol II, which has the desired PGK forward secrecy.

Since the only difference between Protocol II and Protocol I is the final computation of the shared secret K, here we merely present Protocol II graphically in Figure 2.

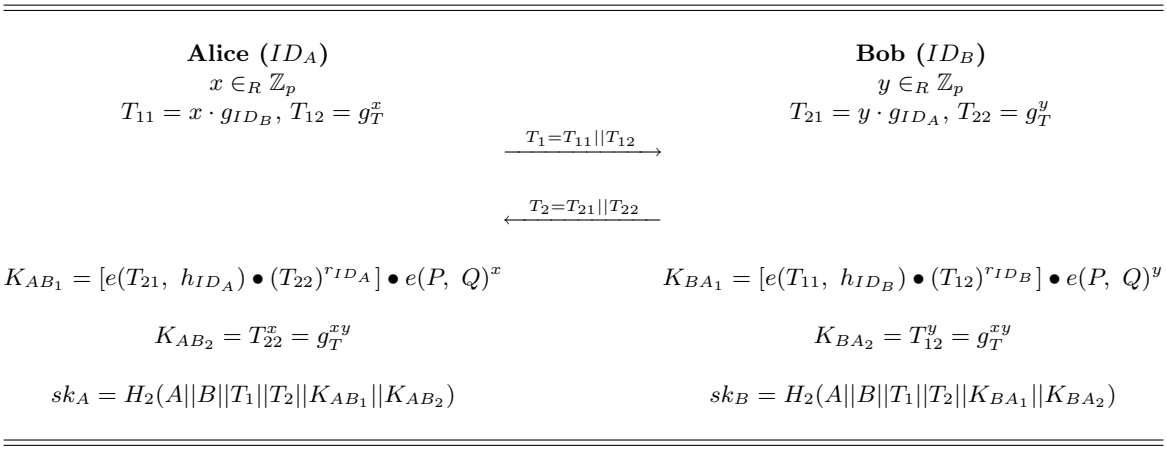

Fig. 2. Protocol II

Remark 4. Protocol II achieves PKG forward secrecy for the fact that the PKG is not able to compute g_T^{xy}.

Using the same ideas from [11] and [19], our protocols can be extended to key agreement between members of distinct domains (i.e., with different PKGs). We leave the details, which are straightforward, to the reader.

6 Conclusion

We have presented two new identity-based authenticated key agreement protocol based on the Gentry's IBE system. The proposed protocols seems to possess all the desired security of a secure key agreement protocol. Our future work is to investigate their formal security proof in an appropriate formal model. Furthermore, we expect that our protocol to be the first secure *identity-based* authenticated key agreement protocols without random oracles.

Acknowledgments

The first author would like to thank Caroline Kudla for her valuable suggestions on designing secure ID-based AK protocols without random oracles, and Raymond Choo for many helpful discussions

on formal security proof for AK protocols. This work was supported in part by the National High Technology Development Program of China under Grant No. 242006AA01Z424 and the National Natural Science Foundation of China under Grant No. 60673079.

References

1. S.S. Al-Riyami and K.G. Paterson. Certificateless public key cryptography. In *Proc. of ASIACRYPT 2003*, LNCS vol. 2894, pp. 452-473, 2003.
2. D. Boneh, M. Franklin. Identity-based encryption from the Weil pairing. In *Proc. of CRYPTO 2001*, LNCS vol. 2139, pp. 213-229. Springer-Verlag, 2001.
3. M. Bellare, and P. Rogaway. Entity authentication and key distribution. In Proc. of CRYPTO 1993, LNCS vol. 773, pp. 110-125. Springer-Verlag, 1994.
4. C. Boyd and R. Choo. Security of two-party identity-based key agreement. In *Proc. of Mycrypt 2005*, LNCS vol. 3715, pp.229-243. Springer-Verlag, 2005.
5. S. Blake-Wilson, D. Johnson, and A. Menezes. Key agreement protocols and their security analysis. In *Proc. of 6th IMA International Conference on Cryptography and Coding*, LNCS vol. 1355, pp. 30-45, 1997.
6. S. Blake-Wilson, A. Menezes. Authenticated Diffie-Hellman key agreement protocols. In *Proc. of SAC 1998*, LNCS vol. 1556, pp. 339-361. Springer-Verlag, 1999.
7. C. Boyd and A. Mathuria. Protocols for Authentication and Key Establishment. Springer-Verlag, June 2003.
8. M. Burmester. On the risk of opening distributed keys. In *Proc. of CRYPTO 1994*, pp. 308-317. LNCS vol. 839, Springer-Verlag, 1994.
9. K.-K. R. Choo, C. Boyd, and Y. Hitchcock. On session key construction in provably secure protocols. In *Proc. of MYCRYPT 2005*, LNCS vol. 3715, , pp. 116-131, Springer-Verlag, 2005.
10. Y. J. Choie, E. Jeong and E. Lee. Efficient identity-based authenticated key agreement protocol from pairings. *Journal of Applied Mathematics and Computation*, 162(1), pp. 179-188, 2005.
11. L. Chen, C. Kudla. Identity based key agreement protocols from pairings. In *Proc. of the 16^{th} IEEE Computer Security Foundations Workshop*, pp. 219-213. IEEE Computer Society, 2002.
12. W. Diffie, M.E. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theory*, 22(6), pp.644 - 654, 1976.
13. T. ElGamal. A public key cryptosystem and signature scheme based on discrete logarithms. *IEEE Trans. Info. Theory*, 31(4), pp. 469-472, 1985.
14. C. Gentry. Certificate-based encryption and the certificate-revocations problem. In *Proc. of EUROCRYPT 2003*, LNCS vol. 2656, pp. 272- 291. Springer-Verlag, 2003.
15. C. Gentry. Practical identity-based encryption without random oracles. In *Proc. of EUROCRYPT 2006*, LNCS vol. 4004, pp. 445-464. Springer-Verlag, 2006.
16. K. C. Goss. Cryptographic method and apparatus for public key exchange with authentication. US Patent 4,956,863, September 1990.
17. C. Kudla. Special signature schemes and key agreement protocols. PhD Thesis, Royal Holloway University of London, 2006.
18. S. Kunz-Jacques and David Pointcheval. About the Security of MTI/C0 and MQV. In *Proc. of SCN 2006*. LNCS vol. 4116, pp. 156C172, 2006.
19. N. McCullagh, P.S.L.M. Barreto. A new two-party identity-based authenticated key agreement. In *Proc. of CT-RSA 2005*, LNCS vol. 3376, pp. 262-274. Springer-Verlag, 2005.
20. A. Menezes, P. van Oorschot and S. Vanstone. Handbook of Applied Cryptography, pp. 237-238. CRC Press, 1997.
21. T. Matsumoto, Y. Takashima and H. Imai. On seeking smart public-key distribution systems. *Trans. IECE of Japan*, E69, pp.99-106, 1986.
22. NIST, SKIPJACK and KEA Algorithm Specification, http://csrc.nist.gov/encryption/skipjack/skipjack.pdf, 1998.
23. E.K. Ryu, E.J. Yoon, and K.Y. Yoo. An efficient ID-based authenticated key agreement protocol from pairings. In *Proc. of NETWORKING 2004*, LNCS vol. 3042, pp. 1458-1463. Springer-Verlag, 2004.
24. A. Shamir. Identity-based cryptosystems and signature schemes. In *Proc. of CRYPTO 1984*, LNCS vol. 196, pp. 47-53. Springer-Verlag, 1984.

25. N. Smart. An ID-based authenticated key agreement protocol based on the Weil pairing. *Electron. Lett.*, 38(13), pp. 630-632, 2002.
26. K. Shim. Efficient ID-based authenticated key agreement protocol based on Weil pairing. *Electron. Lett.*, 39(8), pp. 653-654, 2003.
27. Anoymous Author. A provable ID-based explicit authenticated key agreement protocol without random oracles. Cryptology ePrint Archive, Report 2006/425.
28. S. Wang, Z. Cao. Escrow-free certificate-based authenticated key agreement protocol from Pairings. *Wuhan University Journal of Natural Sciences* (to appear)

A A K-CI Attack on the Protocol from [27]

For completeness, here we first briefly review the protocol from [27], then we introduce our key-compromise impersonation attack on it.

A.1 Review of the Protocol

The protocol in [27] is a 3-pass ID-based AKC protocol, which also composes three stages: `Setup`, `Key Generation` and `Key Agreement`. The `Setup` and `Key Generation` stages are identical to that of Gentry's IBE scheme [15]. So we only describe the `Key Agreement` stage as follows.

Key Agreement [27]. Alice and Bob follow the following steps:

1. Alice picks randomly a $x \in \mathbb{Z}_p$, computes $T_{11} = x \cdot g_{ID_B}$, $T_{12} = g_T^x$. She sends $T_1 = T_{11}||T_{12}$ to Bob.
2. On receiving T_1, Bob
   (a) firstly picks randomly a $y \in \mathbb{Z}_p$, computes $T_{21} = y \cdot g_{ID_A}$, $T_{22} = g_T^y$.
   (b) computes
   $$K_{BA} = [e(T_{11}, h_{ID_B}) \bullet (T_{12})^{r_{ID_B}}]^y = e(P, Q)^{xy}.$$
   (c) computes $T_{23} = H(K_{BA}||T_{11}||T_{12}||T_{21}||T_{22})$ and sends $T_2 = T_{21}||T_{22}||T_{23}$ to Alice.
3. On receiving T_2, Alice do the following:
   (a) computes K_{AB} as follows,
   $$K_{AB} = [e(T_{21}, h_{ID_A}) \bullet (T_{22})^{r_{ID_A}}]^x = e(P, Q)^{xy}.$$
   (b) then computes $V_{T_{23}} = H(K_{AB}||T_{11}||T_{12}||T_{21}||T_{22})$, checks if $T_{23} = V_{T_{23}}$. If it does not hold, Alice rejects and aborts the protocol run. Otherwise, she accepts K_{AB} as the shared session key.
   (c) computes $T_{13} = H(K_{AB}||T_{21}||T_{22}||T_{11}||T_{12})$ and sends $T_3 = T_{13}$ to Bob.
4. Bob computes $V_{T_{13}} = H(K_{BA}||T_{21}||T_{22}||T_{11}||T_{12})$ and checks if $T_{13} = V_{T_{13}}$. If it does not hold, he rejects and aborts the protocol run. Otherwise, he accepts K_{BA} as the shared session key.

In [27], the author proved the above protocol to be secure in a modified formal model of Bellare and Rogaway [3] and claims that the protocol possesses almost all the essential security attributes, including known-key secrecy, impersonation attack resilience, unknown key-share resistance, key-compromise impersonation resilience, perfect forward secrecy and key control resilience. Contrary to the author's claim, however, next we show a K-CI attack on it.

A.2 A Key-Compromise Impersonation Attack

Here we describe our K-CI attack on the protocol from [27]. Assume that Alice's private key $d_{ID_A} =< r_{ID_A}, h_{ID_A} >$ is compromised. Obviously, an adversary Eve who gets this private key can impersonate Alice to any other entity, since d_{ID_A} is the only private key of Alice which exactly identifies her. As previously stated, it is desired that this compromise does not allow the adversary Eve to impersonate other entities to Alice.

In our attacking scenario, the two users Alice and Bob are about to run an instance of the protocol. With the knowledge of Alice's private key d_{ID_A}, Eve tries to impersonate Bob (with identity ID_B) to Alice. The K-CI attack launched by Eve against Alice (and Bob) is described as follows: (Suppose Alice initiates the protocol run intended with Bob by sending out $T_1 = T_{11}||T_{12}$.)

1. After intercepting T_1, Eve
   (a) firstly picks randomly a $y' \in \mathbb{Z}_p$, computes $T_{21} = y' \cdot g_{ID_B}$, $T_{22} = g_T^{y'}$.
   (b) computes
   $$K_{E(B)A} = [e(T_{11}, h_{ID_A}) \bullet (T_{12})^{r_{ID_A}}]^{y'}$$
   (c) computes $T_{23} = H(K_{E(B)A}||T_{11}||T_{12}||T_{21}||T_{22})$ and sends $T_2 = T_{21}||T_{22}||T_{23}$ to Alice.
2. Upon receiving T_2, Alice proceeds as follows:
   (a) compute the session key as
   $$K_{AB} = [e(T_{21}, h_{ID_A}) \bullet (T_{22})^{r_{ID_A}}]^{x}.$$
   (b) then computes $V_{T_{23}} = H(K_{AB}||T_{11}||T_{12}||T_{21}||T_{22})$, checks if $T_{23} = V_{T_{23}}$. If it does not hold, Alice rejects and aborts the protocol run. Otherwise, she accepts K_{AB} as the shared session key.
   (c) computes $T_{13} = H(K_{AB}||T_{21}||T_{22}||T_{11}||T_{12})$ and sends $T_3 = T_{13}$ to Bob.
3. Eve accepts $K_{E(B)A}$ as the shared session key.

Correctness: The above attack is successful, for we have the following equations:

$$\begin{aligned}
K_{E(B)A} &= [e(T_{11}, h_{ID_A}) \bullet (T_{12})^{r_{ID_A}}]^{y'} \\
&= e(y' \cdot T_{11},\ h_{ID_A}) \bullet (T_{12})^{y' r_{ID_A}} \\
&= e(y'(x \cdot g_{ID_B}),\ h_{ID_A}) \bullet (g_T^{x})^{y' r_{ID_A}} \\
&= e(x(y' \cdot g_{ID_B}),\ h_{ID_A}) \bullet (g_T^{y'})^{x r_{ID_A}} \\
&= e(x \cdot T_{21},\ h_{ID_A}) \bullet (T_{22})^{x r_{ID_A}} \\
&= [e(T_{21}, h_{ID_A}) \bullet (T_{22})^{r_{ID_A}}]^{x} \\
&= K_{AB}
\end{aligned}$$

Therefore, Eve can always impersonate Bob (actually, anybody except for Alice) to Alice.

Further Comments. From the keying data generation form, this protocol closely resembles the MTI/C0 protocol [21]. However, as has been showed by Boyd and Mathuria [7], the MTI/C0 protocol is not secure against key-compromise impersonation (K-CI) attack. Interestingly, analogous to the K-CI attack on the MTI/C0 protocol, we find the above K-CI attack on it. This attack reveals the fact that the proof in [27] must be invalid.

Boyd and Choo [4] conjectured that the similarities between many ID-based authenticated key agreement protocols and various protocols using conventional Diffie-Hellman in finite fields may extend to the security properties of these protocols. Our attack here once again justifies their conjecture to some extent.